

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Please cancel claims 1-13 without prejudice. Please add new claims 14-29.

Claims 1-13 (canceled)

14. (new): An array processor comprising:

a physical configuration of at least two processing elements; and

a processor state register storing a context status bit (CSB), the CSB having at least a first state and a second state, each processing element operating to detect the state of the CSB,

the array processor upon detection of the first state of the CSB operating in a first operating context adapted for processing a first software task where the first software task is written for the physical configuration,

the array processor upon detection of the second state of the CSB operating in a second operating context adapted for a second software task where the second software task is written for a second array processor having a different physical configuration.

15. (new): The array processor of claim 14 wherein in the first operating context, the array processor utilizes a first and a second set of register files, the first set of register files associated with one of the processing elements and the second set of register files associated with another of the processing elements.

16. (new): The array processor of claim 15 further comprising:

an eventpoint mechanism to trigger storing the first set of register files in the background while the first software task uses the second set of register files in the foreground.

17. (new): The array processor of claim 15 further comprising:

an eventpoint mechanism to trigger loading the first set of register files in the background while the first software task uses the second set of register files in the foreground.

18. (new): The array processor of claim 14 wherein each processing element of the at least two processing elements has a physical identifier and a virtual identifier, wherein during the processing of the second software task, instructions are loaded to each processing element according to its virtual identifier.

19. (new): A method for providing reconfiguration of a first array processor having a first physical configuration to emulate operation of a second array processor having a second physical configuration, the method comprising:

providing the first array processor having at least two processing elements;

storing a context status bit (CSB), the CSB having at least a first state and a second state;

detecting the state of the CSB;

upon detection of the first state, operating in a first operating context adapted for processing a first software task, the first software task is written for the first physical configuration; and

upon detection of the second state, operating in the second operating context adapted for processing a second software task, the second software task is written for the second physical configuration.

20. (new): The method of claim 19 wherein the operating in the second operating context step further comprises:

setting the CSB to the first state; and

returning the processing to the first operating context.

21. (new): The method of claim 19 wherein the first physical configuration comprises a 1x1 layout and the emulated second physical configuration comprises a 1x0 layout.

22. (new): The method of claim 19 wherein the first physical configuration comprises a 1x2 layout and the emulated second physical configuration comprises a 1x1 layout.

23. (new): The method of claim 19 wherein the first physical configuration comprises a 1x5 layout and the emulated second physical configuration comprises a 2x2 layout.

24. (new): An apparatus for providing efficient sharing of programming resources in a merged very long instruction word (VLIW) sequence processor (SP) and VLIW processor element (PE) processor, the merged VLIW SP/PE processor configurable in a first merged processor configuration or in a second merged processor configuration, the apparatus comprising:

an SP resource file having a first set of registers;

a PE resource file having a second set of registers;

an input for receiving a VLIW presented for execution, the VLIW having at least two instructions, each instruction encoded with a different setting of an SP/PE-bit; and

a processor state register storing context select bit (CSB), the merged VLIW SP/PE processor reading the CSB and the SP/PE-bit of each instruction to select a first merged processor configuration or a second merged processor configuration, the first merged processor configuration adapted for accessing at least one register from the second set of registers when processing an SP instruction.

25. (new): The apparatus of claim 24 wherein the second merged processor configuration adapted for accessing at least one register from the first set of registers when processing an SP instruction and accessing at least one register from the second set of registers when processing a PE instruction.

26. (new): The apparatus of claim 24 wherein the SP resource file is a SP register file, SP address register file, or a SP machine state register file.

27. (new): The apparatus of claim 24 wherein the PE resource file is a PE register file, PE address register file, or a PE machine state register files.

28. (new): The apparatus of claim 24 further comprising:
at least two execution units associated with the at least two instructions in the VLIW; and

a plurality of multiplexers connected to the SP and PE resource files for selecting resource files from which the at least two execution units read data and to which the at least two execution units write data, a portion of the plurality of multiplexers associated with an execution unit controlled by a logical combination of the SP/PE bit and the CSB.

29 (new): The apparatus of claim 24 wherein the VLIW SP processor and VLIW PE processor are indirect VLIW processors.